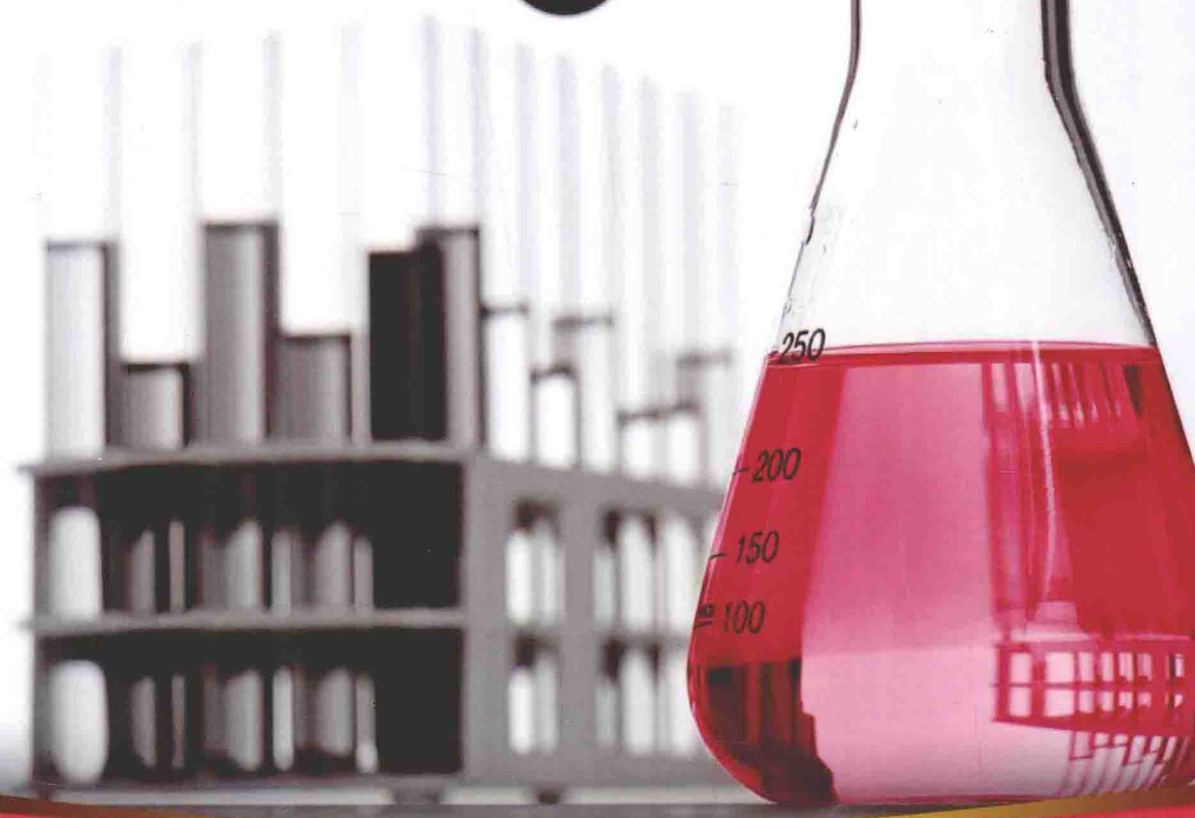


Join the discussion @ p2p.wrox.com



Wrox Programmer to Programmer™



Java Programming Interviews Exposed

Java



程序员成功面试秘籍

[英] Noel Markham 著
郑思遥 译

清华大学出版社

Java 程序员 成功面试秘籍

[英] Noel Markham 著

郑思遥 译

清华大学出版社

北 京

Noel Markham

Java Programming Interviews Exposed

EISBN: 978-1-118-72286-2

Copyright © 2014 by John Wiley & Sons, Inc., Indianapolis, Indiana

All Rights Reserved. This translation published under license.

Trademarks: Wiley and the Wiley logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. Java is a registered trademark of Oracle America, Inc. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

本书中文简体字版由 Wiley Publishing, Inc. 授权清华大学出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2014-3655

Copies of this book sold without a Wiley sticker on the cover are unauthorized and illegal.

本书封面贴有 Wiley 公司防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

Java 程序员成功面试秘籍 (美) 马克·马基斯(Mark Markham) 著; 郑思遥 译. —北京: 清华大学出版社, 2015

书名原文: Java Programming Interviews Exposed

ISBN 978-7-302-39063-

I. ①J… II. ①马… ②郑… III. ①JAVA 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 017216 号

责任编辑: 王 军 刘伟琴

装帧设计: 牛艳敏

责任校对: 成凤进

责任印制: 宋 林

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 三河市君旺印务有限公司

装 订 者: 三河市新茂装订有限公司

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 23 字 数: 560 千字

版 次: 2015 年 3 月第 1 版 印 次: 2015 年 3 月第 1 次印刷

印 数: 1~4000

定 价: 59.80 元

产品编号: 060463-01

译者序

与其说本书是一本面试书，不如说是一本 Java 大全。这本书介绍了 Java 开发所涉及的各个方面的知识，从最基础的语言特性，到设计模式以及 Spring 和 Hibernate 这样的框架；从基本的算法和数据结构，到大型 Java 项目的构建流程；从 Java 虚拟机的基本原理和 Java 并发，到 Android 开发和 Scala 语言的基本思想，无所不包。

如果你是要参加技术面试的候选人，那么可以通过本书对 Java 相关知识查漏补缺，还可以利用本书给出的题目进行演练；如果你是刚接触 Java 的新手，那么可以通过本书迅速了解专业 Java 开发者所需要的知识和素养；如果你是 Java 老手，那么在这本书中一定还能找到自己欠缺的方面。

此外，本书还试图帮助读者建立起测试驱动开发的习惯，在如今大规模使用 Java 开发线上服务的时代，充分的测试能保证你睡个好觉。

当然，我们不能把这本书当成 Java 知识大全，本书以面试题的形式将 Java 相关的很多重要知识点串起来，但如果想要了解更多的细节，还需要参阅相关的文档和涉及相关主题的更深入的书籍。

在本书的翻译过程中，要特别感谢清华大学出版社的编辑们对我的指导和督促，感谢他们对本书翻译稿的修正和润色。还要感谢我的妻子和襁褓中的孩子对我因为翻译本书而对他们关注减少的理解和支持。

本书全部章节由郑思遥翻译，由于时间和水平的限制，翻译稿中难免存在疏漏和错误，敬请广大读者批评指正。

郑思遥

2014 年 12 月

作者简介

Noel Markham 是一名有着将近 15 年 Java 经验的开发者，他涉足的领域包括金融、技术和游戏。最近，他在一家主营社交游戏和数码娱乐的初创公司工作。他面试过从应届毕业生到技术管理者各种层次的开发者。他在英国和海外设立了测评中心以组建完整的开发团队。

贡献者简介

Greg Milette 是一名程序员、作家、咨询师和喜欢实现伟大想法的创业家。从 2009 年他发布了一款名为 Digital Recipe Sidekick 的语音控制菜谱应用以来，他就一直在开发 Android 应用。除了在厨房与他的 Android 设备聊天外，他在 2012 年还与他人合著了图书《Android 传感器高级编程》(清华大学出版社引进并出版)。目前，他是 Gradison Technologies 公司的创始人，他依然在开发伟大的应用。

技术编辑简介

Ivar Abrahamsen 是一名有着 15 年工作经验的软件工程师/架构师，他主要使用 Java 语言，最近在使用 Scala 语言。他在不同国家的大大小小的银行、电信公司和咨询公司工作过。Ivar 目前是伦敦肖尔迪奇区一家初创公司的充满激情的和执着的技术领导者。更多介绍请参见 flurdy.com。

Rheese Burgess 是一名技术领导者，他的工作内容和招兵买马紧密相关。他使用过很多种语言和技术，在社交应用、游戏和金融产品方面有着丰富的经验，横跨初创公司到投资银行的各行各业。

致 谢

感谢 James Summerfield 让我接触到 Wiley 出版公司，从而使这本书有机会出版。

在整个写作过程中，Wiley 的团队给予我很多鼓励和支持：感谢编辑 Ed Connor 和 Robert Elliott，还有很多其他幕后的工作人员。

感谢 Greg Milette 编写了关于 Android 的那一章。

感谢我的同事 Rheese Burgess 和 Ivar Abrahamsen 花时间审读这本书。

感谢我在 Mind Candy 的同事 Sean Parsons、Olivia Goodman 和 Amanda Cowie 给予我的帮助，也感谢我的好朋友 Luca Cannas。

感谢我的家人，谢谢我的父母对我所做的一切事情提供了无尽的帮助和支持。

最后感谢我的妻子 Rebecca，谢谢你在过去几个月能容忍我全身心投入在这本书上，谢谢你给我带来的茶和小点心，谢谢你一直的笑容。没有你就不会有这本书。

前 言

有人会认为面试很难对付，因为面试是一个一对一的考察技术能力的过程。

这本书是准备找工作的 Java 面试者的必备指南。本书提供了足够多的练习材料，让你有足够的信心面对可能会被问到的问题，因此可以帮助你克服 Java 编程面试过程中的恐惧和担忧。

本书介绍的技术

本书是基于 Java SE 7 编写的。如果你是一位有经验的 Java 开发者，但是还没有跟上 Java 7 的步伐，那么可以在本书中学习以下新的语言特性和 API 库。

钻石操作符

编译器会尽可能地推导出泛型实例的类型。因此，我们可以编写 `List<Integer> numbers = new ArrayList<>()` 这样的语句，而不用编写 `List<Integer> numbers = new ArrayList<Integer>()` 这样的语句。这样在使用集合时，特别是在集合嵌套集合时，可以大大减少样板化代码的量。

在 switch 语句中使用 String

最初的 Java switch 语句只能处理数值类型。随着 Java 5 引入了枚举类型，在 switch 语句中还可以使用枚举类型。现在从 Java 7 开始，在 switch 语句中可以使用 String 对象。

新的文件 I/O 库

Java 7 引入了一个新的 I/O 库，关注的是平台无关的非阻塞 I/O。

此外，Java 7 还引入了很多新特性，例如自动资源管理和二进制字面量的表示形式。本书全书使用 Java 7。可以认为，面试官会使用最新的 Java 技术来进行考察，因此随着语言的新版本的发布，你也应该让自己的技能跟上时代的步伐。

有一些 Java 框架和库还没有完全兼容 Java 7，因此在使用具体组件时请查阅最新的文档。

本书的组织结构

全书分为以下 3 个独立的部分。

第 I 部分：面试流程概述

关于面试流程的章节包含两方面的内容：一方面是如何向面试官表现自己，另一方面是一些和 Java 无关但是可能会出现在技术面试中的一般性技术话题。

第 1 章：面试类型分析

该章介绍了雇主的招聘流程中的各个步骤，从电话面试到面对面的技术测试，再到和招聘经理的面谈。

第 2 章：撰写引人入胜的简历

简历及其附信是下一任雇主对你产生的第一印象。该章讨论了如何让你的简历脱颖而出，给出了语言相关的一些小建议和招聘者会关注的一些问题。

第 3 章：技术测试和面试基础知识

任何潜在的雇主都会考察你的技术能力，而且会采用尽可能高效的方式。该章讨论了各种不同类型的编程测试和技术测试，并讨论了如何准备这些类型的测试。

第 4 章：编写核心算法

计算机科学的核心概念是技术测试中经常会考察的内容，其中包含排序和搜索。该章给出了一些不同的搜索和排序算法，并讨论了每一种方法的优劣。

第 5 章：数据结构

在面试中，计算机科学核心概念的相关问题除了排序和搜索之外，常见的主题就是数据的高效存储和表达。第 5 章讨论了列表、树、映射和集合等数据结构，讨论了这些数据结构的表达和使用。

第 6 章：设计模式

该章涵盖了一些面向对象的设计模式，还展示了 Java 库中的类使用的一些模式。

第 7 章：常见面试算法的实现

该章给出了一些常见的面试题，并用 Java 实现了这些题目的解答。本章很多题目都来自于一个很流行的技术面试网站 interviewzen.com。

第 II 部分：核心 Java

这一部分章节的内容是面试官认为有经验的 Java 开发候选人应该了解的内容。

第 8 章：Java 基础

该章覆盖了 Java 的很多语言特性，有经验的 Java 开发者可以利用该章复习基础知识。

第 9 章：基于 JUnit 的测试

本书使用的一个核心思想是单元测试。该章介绍了 JUnit，并讲解了如何通过 JUnit 验证假设和断言。

第 10 章：理解 Java 虚拟机

任何有能力的开发者都会对自己使用的平台有一定的理解，Java 也不例外。该章介绍了 JVM 的一些特性，以及 JVM 和 Java 语言之间的交互。

第 11 章：并发

该章讨论了 Java 的线程模型及使用方法。该章还介绍了 actor 模型，它是并发编程采用的一种新方法。

第 III 部分：组件和框架

这一部分讨论了一些使用 Java 的领域，从数据库到 Web 服务，从流行的框架(例如，Hibernate 和 Spring)到构建和发布企业级应用的工具。在面试某个具体职位时，面试官可能希望你具有这些章节所介绍的一些相关知识，而这些知识通常都会在工作描述中列出，也许你一开始就是因为看到了工作描述才去应聘这份工作。

第 12 章：Java 应用程序和数据库的整合

很多大型 Java 应用程序都会包含数据库组件。该章介绍了数据库操作的标准语言 SQL 以及如何在 Java 中使用 SQL。

第 13 章：创建 Web 应用程序

Java 是创建通过 HTTP 提供数据服务的应用程序的一种常用语言。该章讨论了 3 个流行的框架：Tomcat、Jetty 和 Play。

第 14 章：HTTP 和 REST

该章讨论了 HTTP 的另一种用途：通过 REST (Representational State Transfer)风格创建和使用 Web 服务。

第 15 章：序列化

序列化是传输结构化数据所采用的方法。该章涵盖了 3 种方法：Java 自己的序列化机制以及分别使用 XML 和 JSON 的平台无关的方法。

第 16 章：Spring 框架

Spring 框架是一个比较流行的应用框架，很多雇主在自己的一些甚至全部应用中都使用了这个框架。该章讨论了 Spring 框架的一些组件，包括核心应用上下文、数据库集成以

及集成测试。

第 17 章：使用 Hibernate

Hibernate 是一个将关系数据库数据映射到 Java 对象的框架。该章介绍了 Hibernate，以及如何通过 Hibernate 创建和操作对象。

第 18 章：有用的库

Java 有很多有用且可重用的库。该章介绍了 3 个比较流行的库：Apache Commons、Guava 和 Joda Time。

第 19 章：利用构建工具进行开发

任何大型的 Java 应用程序，特别是涉及多位开发者的应用程序，其构建和打包过程都需要管理良好的流程。该章介绍了构建 Java 应用程序使用的两个最流行的工具：Maven 和 Ant。

第 20 章：Android 开发

最后这一章介绍了 Java 语言的一种现代应用：在 Android 上开发移动应用程序。该章介绍了 Android SDK 的关键组件及其整合方式。

附录 A：Scala 简介

该附录介绍了 Scala 语言，这是一种在 Java 开发团队中越来越流行的语言，因为这种语言使用 JVM 作为平台。该附录介绍了 Scala 语言的一些基本知识、函数式编程的概念以及一些和不可变性相关的约定。

本书读者对象

本书面向的是有一些经验的 Java 开发者：这样的读者应该了解这门语言，而且使用过一段时间，但是对书中有些章节的内容不熟悉甚至完全不了解。如果你从来没有使用过 Java，这本书也会有帮助，特别是第 II 部分中的几章。你应该按顺序阅读这些内容，同时可以参考其他更深入的介绍性材料。

如果你处于面试官的位置，那么这本书也很有用，你可以通过本书找到一些在面试中问问题的灵感。

本书利用的工具

从 <http://www.oracle.com/technetwork/java/javase/downloads/index.html> 可以下载最新的

Java JDK。大部分专业开发者会在集成开发环境(Integrated Development Environment, IDE)中编写 Java 代码。IntelliJ(免费社区版可以从 <http://www.jetbrains.com/idea/download/index.html> 下载)和 Eclipse(可以从 <http://www.eclipse.org/downloads/> 下载)是两个最流行的 IDE。有些面试官在面试过程中可能要求你使用计算机和 IDE 编写代码,因此你也应该熟悉这些 IDE 的基本操作。

下载示例源代码

本书中所有的示例源代码都可以从 www.wiley.com/go/javaprogramminginterviews 下载。下载、编辑和编译这些源代码对你非常有帮助,对于那些你不太熟悉的内容更是如此。另外,<http://www.wrox.com> 和 <http://www.tupwk.com.cn/downpage> 也提供了源代码下载和技术支持。

小结

本书并不是获得 Java 开发者工作的捷径,而是帮助你找到下一份工作的指南。通过本书可以了解面试官在招聘过程中经常会提到的一些话题。

经验对于面试非常有帮助,不仅对于面试的内容本身有帮助,而且也有助于减轻面试本身的陌生感和压迫感。你必须不断地练习面试的技巧,随着参与开发的时间越来越多,你的技能也会不断提高。在面试过程中获得经验会比较困难。每一个面试官都会采用不同的面试方式,整个过程也不会完美。潜在的雇主除了考虑“这位候选人是否足够好”之外还要考虑很多变数,在通盘考虑之后才可能给出录用意向。预算的约束,团队的融合,甚至面试官本身的心情都会对面试造成影响。

如果被拒了,也不要太沮丧,尝试从中吸取一些经验,想一想在以后的面试中会碰到什么问题。

还要记住,面试是一个双向的过程:在面试时要询问关于职位的细节,还要了解未来的同事以及在办公室中的生活情况。如果感觉不对的话,不要害怕拒绝。

祝你在准备面试的过程中好运,也希望你能享受这个过程。祝愿本书能帮你找到理想的工作。

目 录

第 I 部分 面试流程概述

| | |
|-------------------|----|
| 第 1 章 面试类型分析 | 3 |
| 1.1 电话面试流程 | 4 |
| 1.2 技术测试 | 5 |
| 1.3 应对面对面的现场面试 | 7 |
| 1.4 最终的决定 | 8 |
| 1.5 本章小结 | 8 |
| 第 2 章 撰写引人入胜的简历 | 11 |
| 2.1 如何撰写简历和求职信 | 11 |
| 2.2 撰写求职信 | 14 |
| 2.3 本章小结 | 14 |
| 第 3 章 技术测试和面试基础知识 | 17 |
| 3.1 书面技术测试 | 17 |
| 3.2 上机测试 | 18 |
| 3.3 面对面的技术面试 | 19 |
| 3.4 本章小结 | 20 |
| 第 4 章 编写核心算法 | 23 |
| 4.1 关于大 O 符号 | 23 |
| 4.2 列表排序 | 24 |
| 4.3 列表搜索 | 32 |
| 4.4 本章小结 | 34 |
| 第 5 章 数据结构 | 35 |
| 5.1 列表 | 35 |
| 5.2 树 | 39 |
| 5.3 映射 | 45 |
| 5.4 集合 | 48 |
| 5.5 本章小结 | 49 |

| | |
|-----------------|----|
| 第 6 章 设计模式 | 51 |
| 6.1 考察示例模式 | 51 |
| 6.2 常用模式 | 60 |
| 6.3 本章小结 | 64 |
| 第 7 章 常见面试算法的实现 | 67 |
| 7.1 实现 FizzBuzz | 67 |
| 7.2 生成斐波那契数列 | 69 |
| 7.3 实现阶乘 | 73 |
| 7.4 实现库的功能 | 74 |
| 7.5 使用泛型 | 82 |
| 7.6 本章小结 | 85 |

第 II 部分 核心 Java

| | |
|------------------|-----|
| 第 8 章 Java 基础 | 89 |
| 8.1 原始类型 | 90 |
| 8.2 使用对象 | 92 |
| 8.3 Java 数组 | 99 |
| 8.4 String 的使用 | 100 |
| 8.5 理解泛型 | 103 |
| 8.6 自动装箱和拆箱 | 109 |
| 8.7 使用注解 | 110 |
| 8.8 命名约定 | 112 |
| 8.8.1 类 | 113 |
| 8.8.2 变量和方法 | 113 |
| 8.8.3 常量 | 113 |
| 8.9 处理异常 | 113 |
| 8.10 使用 Java 标准库 | 117 |
| 8.11 期待 Java 8 | 120 |
| 8.12 本章小结 | 122 |

| | | | |
|-------------------------------|-----|-------------------------------------|-----|
| 第 9 章 基于 JUnit 的测试 | 123 | 14.4 本章小结 | 232 |
| 9.1 JUnit 测试的生命周期 | 125 | 第 15 章 序列化 | 233 |
| 9.2 使用 JUnit 的最佳实践 | 127 | 15.1 读写 Java 对象 | 233 |
| 9.3 通过 Mock 消除依赖 | 138 | 15.2 使用 XML | 236 |
| 9.4 通过行为驱动的开发进行 系统测试 | 144 | 15.3 JSON | 242 |
| 9.5 本章小结 | 146 | 15.4 本章小结 | 245 |
| 第 10 章 理解 Java 虚拟机 | 149 | 第 16 章 Spring 框架 | 247 |
| 10.1 垃圾回收 | 149 | 16.1 Spring 核心及应用上下文 | 247 |
| 10.2 内存调优 | 151 | 16.2 Spring JDBC | 257 |
| 10.3 JVM 和 Java 语言之间的 互操作性 | 154 | 16.3 集成测试 | 261 |
| 10.4 本章小结 | 159 | 16.4 Spring MVC | 264 |
| 第 11 章 并发 | 161 | 16.5 本章小结 | 271 |
| 11.1 使用线程 | 161 | 第 17 章 使用 Hibernate | 273 |
| 11.2 使用并发 | 167 | 17.1 使用 Hibernate | 273 |
| 11.3 actor | 171 | 17.2 本章小结 | 286 |
| 11.4 本章小结 | 176 | 第 18 章 有用的库 | 289 |
| 第 III 部分 组件和框架 | | 18.1 通过 Apache Commons 去除 样板化的代码 | 289 |
| 第 12 章 Java 应用程序和数据库的 整合 | 179 | 18.2 利用 Guava 集合进行开发 | 292 |
| 12.1 SQL 简介 | 179 | 18.3 使用 Joda Time 库 | 298 |
| 12.2 JDBC: 整合 Java 和数据库 | 193 | 18.4 本章小结 | 302 |
| 12.3 利用内存数据库进行测试 | 199 | 第 19 章 利用构建工具进行开发 | 303 |
| 12.4 本章小结 | 200 | 19.1 通过 Maven 构建应用程序 | 303 |
| 第 13 章 创建 Web 应用程序 | 203 | 19.2 Ant | 311 |
| 13.1 Tomcat 和 Servlet API | 203 | 19.3 本章小结 | 313 |
| 13.2 Jetty | 209 | 第 20 章 Android 开发 | 315 |
| 13.3 Play 框架 | 215 | 20.1 基础知识 | 316 |
| 13.4 本章小结 | 220 | 20.1.1 组件 | 316 |
| 第 14 章 HTTP 和 REST | 223 | 20.1.2 Intent | 317 |
| 14.1 HTTP 方法 | 223 | 20.1.3 Activity | 320 |
| 14.2 HTTP 客户端 | 226 | 20.1.4 BroadcastReceiver | 323 |
| 14.3 通过 REST 创建 HTTP 服务 | 228 | 20.1.5 Service | 324 |
| | | 20.2 用户界面 | 328 |
| | | 20.3 持久化 | 336 |
| | | 20.4 Android 硬件 | 339 |
| | | 20.5 本章小结 | 343 |
| | | 附录 A Scala 简介 | 345 |

第 I 部分

面试流程概述

第 I 部分讲解技术开发者面试中的基本要素，覆盖了从简历撰写到技术相关话题的内容。这一部分的内容不涉及具体的编程语言。

- 第 1 章讨论用于考查候选人水平的各种类型的面试。
- 第 2 章讨论如何编写合适的简历和求职信，并且指出了面试官在阅读简历的时候通常会关注哪些内容。
- 第 3 章描述如何准备各种不同类型的面试。
- 第 4 章讲解一些基本的算法，例如排序和搜索算法。这些算法不仅是面试中很喜欢考察的内容，而且也是在日常编程工作中必须理解的内容。
- 第 5 章介绍一些重要的数据结构。这些数据结构不仅经常出现在面试中，而且在大部分开发工作中这些数据结构的知识也是不可或缺的。
- 第 6 章讨论设计模式。设计模式也是面试中的常见主题。
- 第 7 章讨论一些有用的面试题，这些题目可能会出现在各种编程语言的面试中，不仅限于 Java 语言的面试。

第 1 章

面试类型分析

简单地说，雇主张罗面试的原因要么是为了招揽新的人才，要么是为了填补团队的空缺以提升生产力。在一个开发团队或部门中，你会发现人们具有各种各样不同的技能，这些不同的技能正是一个团队凝聚力的关键。一个人单枪匹马不太可能开发并管理一个专业水准的应用软件，这里面有太多的事情需要处理：需要为产品的客户开发所要求的特性，需要维护测试环境，需要响应运营团队随时可能提出的请求，还需要应付开发团队的日常管理事务。开发团队通常需要一名或多名应用程序开发人员。甚至开发人员也有具体的职责，例如有专门负责前端开发的开发者，还有专门负责数据库开发的开发者。有的团队比较走运，甚至还有专门的构建(build)经理。

不同等级的经验值也很重要。有的开发者参与过多个应用软件，并给这些应用提供支持，这些开发者是每一个新项目成功的关键。而毕业生和只有几年工作经验的开发者也很重要：这些雇员对应用程序的开发往往会有不同的视角和方法。在由多人组成的大型团队中，有经验的成员可以在开发大型企业级应用程序的过程中对新手进行指导和训练。

在面试新人才时，不同公司之间的面试官采用的面试流程以及面试官本人的经验可能存在很大的差别。对新开发者的面试往往由团队中的开发者负责完成，这是情理之中的事情。首先，也是最重要的，对于一名开发者来说，日常工作是要编写和发布经过测试的且可用的应用软件，而不是对别人进行面试。基于上述原因，对别人进行面试是开发者的额外工作，因此开发者有可能会准备不充分，甚至有可能对面试根本没有兴趣。对于现场面试来说，这是第一个要克服的困难：如何让面试官对你产生兴趣。

各大公司，特别是高科技公司和数码公司，越来越能够意识到招聘流程的重要性。有一些更为激进的公司甚至会将招聘工作放进公司的内部目标，重视程度可见一斑。雇主有责任将自己的公司打成一个有吸引力的工作场所，以吸引顶尖的人才，从而带来更高的生产力、更为成功的业绩以及更高的盈利能力。

对于面试者来说，简历(resume 或 Curriculum Vitae(CV))产生的是第一印象。下一章会

讨论如何撰写吸引眼球的简历。

随着你在面试某个职位或某家公司的过程中逐步推进，你会遇到不同的面试风格和面试方法。一般情况下，雇主在为候选人设计“面试流水线”时会尽可能地考虑高效率，面对面的现场面试会安排在整个面试流程的后期进行。

面试流水线中的第一个步骤通常是电话面试(phone screening)，然后是一轮或多轮技术性更强的测试，最后是几轮面对面的面试。

1.1 电话面试流程

在各个公司的面试流程中，第一轮面试通常是电话面试。从公司的角度来说，这种安排是有好处的，因为安排当面的现场面试需要大量的人力：需要为所有面试者、HR(人力资源)安排时间，有些情况下还需要安排招聘小组和会议室等资源。这种安排对应聘者来说也是有好处的，因为不需要暂离当前工作太长时间。很多人不喜欢在找工作时被当前雇主发现，因此在办公场所找一个安全的角落或找一个会议室参加电话面试应该不会太困难。

在进行电话面试时，一定要在面试之前做好充足的准备。如果需要在工作时接这个电话，那么请事先预定好一个会议室或在工作场所中找到一个安静的角落。如果都不现实的话，可以找一家咖啡馆。一定要事先确定好周遭环境的噪声水平在可以接受的范围内：你肯定不希望被咖啡馆内店员穿堂的喊声或巨大的背景音乐所分心。

如果需要远程现场编码测试，一定要确保你所在的地方有互联网连接。手边要准备好笔和纸，因为在通话过程中有可能要记录一些之后要问的问题。在输入代码时使用免提设备，可以极大地提升面试者的通话质量。不要浪费宝贵的时间不断地重复问题或重复解释自己的答案，这对于面试双方来说都是很愉快的。

可以在手边准备一些关于可能会在通话中讨论的话题的材料。这并不是作弊。准备了这些材料有助于安抚你紧张的神经，这种需要通过电话向从未谋面的陌生人讲解技术问题的体验会让人感到不舒服和不自然，因此事先准备好一些材料可以帮助快速适应这样的场合。不过要注意的是，在和团队进行面对面的现场面试时，是不可以带任何这一类材料的。

电话面试通常会持续 30~60 分钟，在这个过程中可能会遇到一些非常高层次的问题。在电话面试中，这一类问题通常都会涉及语言无关的算法问题。面试官有可能要求你口头描述这些算法，甚至有可能要求你通过某个共享的协作式文档编辑器(例如 Google Docs)或某个专门为面试设计的网站(例如 Interview Zen (www.interviewzen.com))进行现场编码。

在面试流程中的任何时刻，你都应该对你所应聘的公司有一些基本的了解。在公司网站上都有“About Us(关于我们)”页面。可以阅读他们的博客页面，寻找他们是否有微博账号。如果应聘的是一家小型公司或初创公司，试着多查找一些关于公司高管的资料，例如 CEO(首席执行官)和 CTO(首席技术官)的资料。这些高管可能会在当地的开发社群中非常活跃。

在面试的最后，面试官通常都会问你是否还有问题要问。如果没有问题可问，会留下